Examination Computational Statistics

Linköpings Universitet, IDA, Statistik

Course code and name:	732A90 Computational Statistics
Date:	2019/05/15, 8–13
Assisting teacher:	Krzysztof Bartoszek
Allowed aids:	Printed books, 100 page computer document, and material in the zip file **extra_material.zip**
Grades:	A= $[18-20]$ points B= $[16-18)$ points C= $[14-16)$ points D= $[12-14)$ points E= $[10-12)$ points F= $[0-10)$ points
Instructions:	Provide a detailed report that includes plots, conclusions and interpretations. If you are unable to include a plot in your solution file clearly indicate the section of R code that generates it. Give motivated answers to the questions. If an answer is not motivated, the points are reduced. Provide all necessary codes in an appendix. In a number of questions you are asked to do plots. Make sure that they are informative, have correctly labelled axes, informative axes limits and are correctly described. Points may be deducted for poorly done graphs. Name your solution files as: **[your anonymous exam account]_[own file description].[format]** If you have problems with creating a pdf you may submit your solutions in text files with unambiguous references to graphics and code that are saved in separate files There are **TWO** assignments (with sub–questions) to solve. Provide a separate solution file for each assignment. Include all R code that was used to obtain your answers in your solution files. Make sure it is clear which code section corresponds to which question.

NOTE: If you fail to do a part on which subsequent question(s) depend on describe (maybe using dummy data, partial code e.t.c.) how you would do them given you had done that part. You *might* be eligible for partial points.

Assignment 1 (10p)

In statistics it is often very important to be able to calculate the sample variance. However, as this is a computational procedure it is subject to potential numerical problems. For this exercise assume that sample data comes from a normal distribution, $\mathcal{N}(\mu, \sigma^2)$ and you are allowed to use `rnorm()` if needed. The observed sample will be denoted $\{x_i\}_{i=1}^n$ and by $\overline{x}_n$ we denote the sample average.

Question 1.1 (6p)

You should be familiar with two formulae for calculating the sample variance:

$$S_1^2 = \frac{1}{n-1}\sum_{i=1}^{n}(x_i - \overline{x})^2 \tag{1}$$

and

$$S_2^2 = \frac{n}{n-1}\left(\frac{1}{n}\sum_{i=1}^{n} x_i^2 - (\overline{x})^2\right). \tag{2}$$

For what magnitudes of μ and σ^2 can S_1^2 and S_2^2 be numerically problematic? Motivate why and illustrate this by calculating the values of S_1^2 and S_2^2 for different, appropriately chosen, values of μ, σ^2 using simulated data. Provide plots (e.g. of the error in estimating σ^2) and comments explaining what is happening with the variance calculations.

Question 1.2 (4p)

Another way to calculate the variance is using the Youngs and Cramer algorithm. This algorithm

Algorithm 1 Youngs and Cramer variance calculation algorithm

$T = x_1$
$RSS = 0$
for $j = 2 \ldots n$ **do**
  $T := T + x_j$
  $RSS = RSS + \frac{(j \cdot x_j - T)^2}{j(j-1)}$
end for
return $RSS/(n-1)$

is provided for you as the function `var_YC()` in the file `YoungsCramer.R`. Compare with S_1^2 and S_2^2 how well it estimates the variance parameter. Try it out on the problematic regions for S_1^2 and S_2^2. Provide plots and comments. How well does it recover the true value of σ^2? Provide a graph of the error in σ^2's estimation for all three methods for an illustrative (i.e. that one can observe where a given method is better or worse) range of σ^2 values. Provide comments.

Assignment 2 (10p)

In a certain currency there are coins of denomination 1, 2, 5, 10 and 20. They weight $\pi/3$g, 2πg, πg, 5πg and 4.5πg respectively. Your task is to design an optimization function that given an amount chooses the coin composition that equals this amount and tries to minimize both the total number of coins and the total weight of the coins.

Question 2.1 (2p)

Propose and write down an objective function for the problem. Provide motivations for it.

Question 2.2 (6p)

Propose and implement an optimization method. Provide some short motivations for your choice of method and problem specific implementation decisions.

Question 2.2 (2p)

Run your optimization for amounts 10 and 150. Provide some plots concerning your optimization method, e.g. its path to the solution.

TIP: You are allowed to use `optim()`, but this might not be required. In `R` you obtain the value of π through the constant `pi`.

TIP: Depending on your objective function, choice of method and implementation it might or might not be possible to have the correct amount of money. If you cannot guarantee this, then your method should have an amount as close as possible to the correct amount. A solution that guarantees the correct amount will be scored higher than one which guarantees not less than, which in turn is better than one that just tries to be as close as possible.

COMMENT: You might recall that if the goal was only to minimize the amount of coins, then there is a deterministic algorithm solving this exactly. Furthermore, if the weights were integers, then the problem could be stated as an integer linear programming problem. None of these two are directly related to the possible optimization methods. In principle they could be used to provide starting points, but the time required to implement these sub–procedures could be prohibitive.